Detection and Defense Approaches for Sensitive Data Leakage in Cloud Computing

Rishti Gupta, Arizona StateUniversity
rgupta75@asu.edu, 1217211814

***Abstract*— The growth of modern technologies and an extensive amount of data that is generated daily has made cloud one of the major resources used by both private organizations and the public users who use it to store and retrieve the data. The project emphasizes on the data leakage which is a threat that can eventually cause major problems like data theft, security breach, identification theft, unauthorized access etc. Thus, it is very vital to have a proper defense mechanism against the data leakage and also crucial to detect the data leakage as soon as it happens so that proper actions can be taken to retrieve the leaked data and prevent any further loss. In this project, various research papers related to data leakage in cloud computing have been extensively studied and logical findings for the same have been presented in the later sections.**

I. INTRODUCTION

Cloud computing is an innovation methodology that is based on the principle of pay per utilize access. It enables gathering of shared resources for stockpiling, servers, specific systems, administration and applications, with no physical access to the cloud. There are a number of businesses which use this facility to improve their efficiency by transferring data from one location to another, for example, storing data related to keeping money, instruction and social insurance.

The primary objective of information security is not to share the data to unapproved elements. In any case, it is constantly unconceivable to forestall information spillage/leakage since there is a need to access and offer the information to utilize that data. Typically, information spillage happens while getting, modifying and sharing the data. Business will get affected when sensitive data, for example, client subtleties and exchange subtleties is leaked to the individuals who are in the same business.

Recently, more dedication is given towards development of different security frameworks and mechanisms for user data protection. Although there are different aspects of ensuring security of systems, data leakage issues are often ignored and more focus is laid on security enhancement. The main goal of our project is to dig into the crucial security issues caused due to data breaches and the significance of data leakage in the se.

Since a lot of data, sensitive or otherwise, is being stored on the cloud, detection and prevention of various attacks on the data has become an important aspect to be considered for data security. The project aims at analyzing the data leakages in cloud computing motivated by recent data breaches that happened at Facebook, iCloud, Sony, etc.

II. SYSTEM GOALS AND SCOPE

In this project, different data leakage detection and defensive mechanisms have been identified and classified based on their characteristics in different layers of the cloud system. The project aims to evaluate the various defense techniques and provide a comparative analysis. Any potential solutions to the unresolved drawbacks to data security in cloud computing systems have also been discussed. Two main topics discussed in the project are:

- **Detection of Data Leakage:** Data leakage is defined as the accidental or intentional distribution of private organizational data to unauthorized entities. A breach in security can lead to accidental or unlawful destruction, loss, alteration, unauthorized disclosure of, or access

to, personal data transmitted, stored or otherwise processed. Under this section, we discuss and analyze data leakage attacks classifying them based on several aspects.

- **Defense Approaches:** There are many types of currently known data leakage problems. We are working towards understanding and justifying the existing mechanisms which are used to mitigate the data leakage problem. Although we have many advanced mechanisms currently employed to mitigate the data leakage problems, there still exist several critical problems which aren't tackled efficiently by these mechanisms.

III. OVERVIEW

The project as stated above discusses techniques to evaluate the detection of data leakage and various defense mechanisms for preventing the data leakage of highly sensitive and confidential data.

A. Evaluation of Detection Techniques:

- **Detection using Tracing algorithms:**
  The algorithm is used to detect the operations such as copy, rename and file movement within a local machine in a cloud as well as it is used to detect file movement across different machines in the same cloud. It helps in achieving transparency and accountability in the cloud computing environments. The drawback of this algorithm was that it was unable to detect if a new file was created in which content from an existing file was copied, and was unable to distinguish if unrelated files were read and created in a sequence thus leading to false positives.

- **Detection using DWT:**
  The method tackles attacks like Gaussian noise attack, Gaussian blur attack, Salt & pepper noise attack, and Speckle noise attack efficiently. It works well for keeping the ownership of image data secure. But, this scheme is unable to provide a method if the intruder only plans on using the information extracted from the leaked data just to gain knowledge.

- **Dynamic Data Leakage Detection Model Based Approach:**
  This technique makes the data more secure in the mapreduce framework by notifying when a data leakage occurs. The results of the algorithm show that the processing time of the data has improved on the basis of distribution of equal quantity of data and also approximately equal response time from each Virtual Machine. The overall performance of the system is improved by reducing the data by approximately 70-80% by using map reduce function. The algorithm has also improved the probability of finding a guilty agent by using the s-max algorithm.

- **Detection and prevention using MyDLP:**
  MyDLP is an open source data loss prevention software. It runs with multi-site configurations on network servers and endpoint computers. It allows the user to monitor, inspect and prevent all outgoing confidential data from the organization in use. This is done through accurate pattern matching detectors called content blades which identifies sensitive data using deep content analysis. It is also used to prevent data leakage as it blocks all outgoing confidential data.

B. Evaluation of Defense Techniques:

- **Defense using MetaData Based Data Storage Mechanism:**
  Pros: This method proposes a new methodology that aims at making the data invaluable to the hacker rather than concentrating on restricting the hacker. It makes data valuable only during acquisition and while the data is being updated.
  Cons:The main drawback is that the DME in the model has to be initially configured which takes a lot of effort and then migration of the existing data to the new model.

- **Defense using Multi-tier Security Approach:**
  The technique improves the security by using a 3-level authentication. The proposed algorithm outperforms the existing techniques w.r.t. Data Center Processing Time, Total Virtual Machine Cost etc.

- **Defence using Fog Computing:**
  Pros: The method combines user behavioural profiling and launching disinformation attacks using decoy files on a suspected intruder. It has a high probability to detect a malicious intruder and protect the real users data from insider attacks.
  Cons: A real user can unintentionally open a decoy file which can trigger the disinformation attack. Also, over the course of time, the machine learning model's accuracy of classifying intruders access patterns might decline because the real user's access pattern becomes increasingly complex.

- **Comparison of Encryption algorithms:**
  Encryption is an important technique for protection of data from unauthorized access. The most commonly used algorithms used for encryption and decryption are RSA, AES, DES and Blowfish. The execution time required for the RSA algorithm is highest. Also, the space required for this algorithm is large compared to other modern algorithms. A public key is used for RSA, which causes lower security of data and thus providing security only to users and not the service provider. Due to these disadvantages, RSA is suitable only when dealing with small data. Another algorithm used is the Blowfish algorithm. It takes less execution time compared to RSA. It requires the least space compared to other algorithms and requires less than 5kb for execution.

IV. INDIVIDUAL CONTRIBUTION

I have researched various detection and defense mechanisms for the data leakage attacks in cloud computing for the project.
In the early phase, I was involved in working on detection of the attack and discussions related to it. I successfully completed the basic concepts of data leakage, its causes and different techniques to mitigate/ prevent the data leakage.
In the later phase, I worked with other team members on various defense mechanisms and to analyze which techniques lead to better results. My research was related to the MetaData based Data Storage Model which is an approach to make the data invaluable while the data is stored. Some important aspects of the mechanism are discussed below:

Metadata Based Data Storage Model:
This methodology ensures that data is invaluable during static residence and gains value only during acquisition or while being updated. The solution will secure data during its time it is residing in the storage location. This method ensures that any information is of use only when the fragments of the information are available and are mapped/related to each other.

Design of the model:
In the model, data is divided into Public Data Segment (PDS) and Sensitive Data Segment (SDS). SDS is further segregated into smaller units until each fragment doesn't have any value individually. The fragmentation need not be of multiple levels. Instead, an effort has to be put in to identify the key element that makes the data sensitive and should be fragmented separately. Figure 1 demonstrates this data fragmentation:

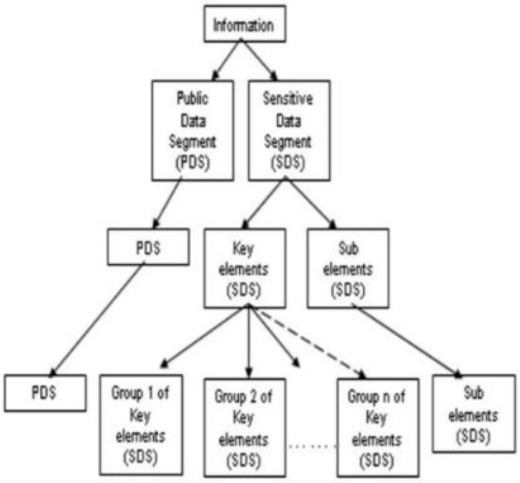

Fig. 1. Data fragmentation

The mapping required to re-assemble the data should be conducted simultaneously while this fragmentation is done.

Methodology:
A banking database which stores user data along with credit card information was considered. The schema is of the form of tables. Some tables contain personal user information and some tables store information regarding credit cards and will be mapped using their ids. The entire model is migrated to the proposed model via the DME. The user needs to provide schema information to the DME and also supply the metadata. The paper considers three categories of metadata in this example:

1. The data which is having low value is considered as 'Normal'.
2. The data which is having high value is considered as 'Critical'
3. The data which has value when mapped with other data is considered sensitive. And the data that maps 'Sensitive' or 'Critical' data to 'Normal' data is also considered 'Sensitive'.

The following tables explain the architecture of the database:

Table	**Metadata**
Customer	Normal
Membership	Sensitive
Credit Card	Critical
Customer_Creditcard	Sensitive

Table I: Metadata Information

Table II lists the metadata of the database in which the data falls under the 'sensitive' category and depicts the current situation:

Table	**Metadata**
Customer	Normal
Membership	Sensitive
DME_Creditcard	Sensitive_DME
Customer_Creditcard	Sensitive
DME_Creditcard_Sensitive	Sensitive_DME
DME_Creditcard_Mapper	Sensitive_DME

Table II: Metadata Information After Fragmentation

After fragmentation is completed, the DME segregates the schema, separating out the data modified by DME, 'Originally Sensitive' data and 'Normal' data is shown in Table III.

Normal	Originally Sensitive	Sensitive DME
Customer	Membership	DME_Creditcard
	Customer_Creditcard	DME_Creditcard_Sensitive
		DME_Creditcard_Mapper

Table III: Segregated Schema

An extra mapping is required to map the original mapping table with the fragmented units. This is stored in a separate table.

Original Table Name	**New Table Name**
Creditcard	DME_Creditcard
Creditcard	DME_Creditcard_Sensitive
Creditcard	DME_Creditcard_Mapper

Table IV: DME_Mapper Table

Now each database contains data that does not have value in itself. An intruder who gets access to the data during the static phase throughout the data can not use the data to exploit the information in any way.

V. LESSONS LEARNT

As a deputy leader, I assisted the leader in conducting the executive tasks of the group including the minutes of the meeting, group meetings and motivating the team to work together to complete the required tasks on time. I have always taken the responsibility and have put in a constant amount of efforts to maintain the team spirit and work together to achieve the desired goal. Thus, the project has helped me to learn to work as a team and I believe that this is an important skill to be gained to be ready to work in the corporate industry. As a team member, I have researched various detection and defense mechanisms for the data leakage attacks in cloud computing for the project. I was involved in working on detection of the attack and discussions related to it. I successfully acquired knowledge about the basic concepts of data leakage, its causes and different techniques to mitigate the data leakage.

VI. TEAM MEMBERS

Srinivasula Reddy Kalluri (Leader), Rishti Gupta (Deputy leader), Sai Hemanth, Krishna Sumanth Gundluru,Vishwarajsinh Sodha, Nachiketa Pathak, Akshay Agarwal, Venkatraman Balasubramanian, Avinash Khatwani, and Pranjali Jagtap

VII. CONCLUSION

As the usage of cloud is increasing, its security has become a vital issue. The various security challenges related to cloud infrastructure are confidentiality, integrity, availability and privacy issues. Different detection and defense approaches have been evaluated in the research. Hybrid solutions are highly recommended depending upon the application..

VIII. ACKNOWLEDGEMENT

I am grateful to Prof. Stephen S Yau for his constant encouragement and support towards the project. I would also humbly extend my thanks to all the team members whose constant hard-work and persistence brought a successful end to the project.

REFERENCES

[1] Bollam, N., & Malsoru, M. V. (2011). Review on Data Leakage Detection. International Journal of Engineering Research and Applications (IJERA), 1(3), 1088-1091.

[2] Subashini, S., & Kavitha, V. (2011, October). A metadata based storage model for securing data in cloud environments. In 2011 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (pp. 429-434). IEEE.

[3] Shabtai, A., Elovici, Y., & Rokach, L. (2012). A survey of data leakage detection and prevention solutions. Springer Science & Business Media.

[4] Carlson, F. R. (2014). Security analysis of cloud computing. arXiv preprint arXiv:1404.6849.

[5] Indira, B., & CH, D. M. A LITERATURE REVIEW ON DATA LEAKAGE DETECTION IN CLOUD COMPUTING.